

AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1 1. (Original) An authenticated-encryption method that uses a key, a nonce
2 and an n-bit block cipher to encrypt a message into a ciphertext, the method
3 comprising:
4 partitioning the message into a message body comprising a sequence of n-
5 bit message blocks, and a message fragment of at most n bits;
6 generating a sequence of offsets from the nonce and the key;
7 computing a ciphertext body using the block cipher, the message body, the
8 key, the nonce, and the sequence of offsets;
9 computing a ciphertext fragment using the block cipher, the message
10 fragment, the key, and an offset;
11 computing a tag as a function of the message body, the message fragment,
12 the sequence of offsets, and the key; and
13 defining the ciphertext to include the ciphertext body, the ciphertext
14 fragment, and the tag.

1 2. (Original) The method of claim 1, wherein generating the sequence of
2 offsets involves:
3 determining a first offset as a function of the nonce and the key; and
4 determining each subsequent offset by combining a previous offset and a
5 basis offset, wherein each basis offset is determined as a function of the key.

1 3. (Original) The method of claim 1, wherein generating the sequence of
2 offsets involves determining an offset by combining a base offset and a fixed
3 offset, wherein the base offset is a function to the key and the nonce, and the fixed
4 offset is a function of the key and the position of the offset in a sequence of
5 offsets.

1 4. (Original) The method of claim 1, wherein generating the sequence of
2 offsets involves:
3 generating a sequence of fixed offsets from the key;
4 generating a base offset from the key and the nonce;
5 generating a sequence of translated offsets by combining each fixed offset
6 with the base offset to get a corresponding translated offset; and
7 using the sequence of translated offsets as the sequence of offsets.

1 5. (Original) The method of claim 4, wherein the key determines a
2 sequence of basis offsets and each fixed offset is determined by xoring some
3 combination of basis offsets.

1 6. (Original) The method of claim 5, wherein each basis offset except for
2 the first basis offset is determined by a shift and a conditional xor applied to a
3 previous basis offset.

1 7. (Original) The method of claim 5, wherein the order that basis offsets
2 are combined into fixed offsets is determined according to a Gray code.

1 8. (Original) The method of claim 1, wherein generating the sequence of
2 offsets involves:
3 computing a sequence of basis offsets from the key;

4 computing a base offset from the key and the nonce; and
5 computing the first offset in the sequence of offsets as a function of the
6 base offset, the key, and the nonce, and computing each subsequent offset in the
7 sequence of offsets by combining the prior offset with a basis offset.

1 9. (Original) The method of claim 1, wherein generating the sequence of
2 offsets involves:
3 computing a key-variant by enciphering a constant with the block cipher,
4 wherein the block cipher is keyed by the given key; and
5 computing the sequence of offsets as a function of the key variant and the nonce.

1 10. (Original) The method of claim 1, wherein computing the ciphertext
2 body involves:
3 combining each message block in the message body with a corresponding
4 offset to produce a corresponding input block;
5 applying the block cipher to each input block to produce a corresponding
6 output block;
7 combining each output block with a corresponding offset to produce a
8 corresponding ciphertext block; and
9 concatenating the ciphertext blocks to determine the ciphertext body.

1 11. (Original) The method of claim 1, wherein computing the ciphertext
2 fragment involves:
3 computing a precursor pad as a function of an offset and the length of the
4 message;
5 computing a pad by applying the block cipher to the precursor pad; and
6 computing the ciphertext fragment by combining the message fragment
7 and the pad.

1 12. (Original) The method of claim 1, wherein computing the tag involves:
2 computing a checksum as a function of the message, the ciphertext
3 fragment, and the sequence of offsets; and
4 computing the tag as a function of the checksum, the key, and an offset.

1 13. (Original) The method of claim 1, wherein computing the tag involves:
2 computing a checksum from at least the message ;
3 combining the checksum with an offset to produce a precursor full tag;
4 computing a full tag by applying the block cipher to the precursor full tag;
5 and
6 computing a tag as a portion of the full tag.

1 14. (Original) An authenticated-encryption method that uses a key, a
2 nonce, and an n-bit block cipher to decrypt a ciphertext into a message or a
3 message-invalid signal, the method comprising:
4 partitioning the ciphertext into a ciphertext body comprising a sequence of
5 n-bit ciphertext blocks, a ciphertext fragment of at most n bits, and a tag;
6 generating a sequence of offsets from the nonce and the key;
7 computing a message body using the block cipher, the ciphertext body, the
8 key, the nonce, and the sequence of offsets;
9 computing a message fragment using the block cipher, the ciphertext
10 fragment, the key, and an offset;
11 computing a new tag as a function of the message body, the message
12 fragment, the sequence of offsets, the block cipher, and the key; and
13 comparing the new tag with the tag;
14 if the new tag matches the tag, returning the message, wherein the message
15 includes the message body and the message fragment; and
16 if the new tag does not match the tag, returning a message-invalid signal.

1 15. (Original) The method of claim 14, wherein generating the sequence of
2 offsets involves:

3 generating a sequence of fixed offsets from the key;
4 generating a base offset from the key and the nonce;
5 generating a sequence of translated offsets by combining each fixed offset
6 with the base offset to get a corresponding translated offset; and
7 using the sequence of translated offsets as the sequence of offsets.

1 16. (Original) The method of claim 14, wherein computing the message
2 body involves:

3 combining each ciphertext block in the ciphertext body with a
4 corresponding offset to produce a corresponding output block;
5 applying the block-cipher inverse to each output block to produce a
6 corresponding input block;
7 combining each input block with a corresponding offset to produce a
8 corresponding message block; and
9 defining the message body to be the sequence of message blocks.

1 17. (Original) The method of claim 14, wherein computing the message
2 fragment involves:

3 computing a precursor pad as a function of an offset and the length of the
4 ciphertext;
5 computing a pad by applying the block cipher to the precursor pad; and
6 computing the message fragment by combining the ciphertext fragment
7 and the pad.

1 18. (Original) The method of claim 14, wherein computing the tag
2 involves:

3 computing a checksum as a function of at least the message; and
4 computing the tag as a function of the checksum, the key, and an offset.

1 19. (Original) A computer-readable storage medium storing instructions
2 that when executed by a computer cause the computer to perform an
3 authenticated-encryption method that uses a key and a nonce to encrypt a message
4 into a ciphertext, the method comprising:
5 partitioning the message into a message body including a sequence of n-bit
6 message blocks, and a message fragment of at most n bits;
7 generating a sequence of offsets from the nonce and the key;
8 computing a ciphertext body using a block cipher, the message body, the
9 key, the nonce, and the sequence of offsets;
10 computing a ciphertext fragment using the block cipher, the message
11 fragment, the key, and an offset;
12 computing a tag as a function of the message body, the message fragment,
13 the sequence of offsets, and the key; and
14 defining the ciphertext to include the ciphertext body, the ciphertext
15 fragment, and the tag.

1 20. (Original) The computer-readable storage medium of claim 19,
2 wherein generating the sequence of offsets involves:
3 determining a first offset as a function of the nonce and the key; and
4 determining each subsequent offset by combining a previous offset and a
5 basis offset, wherein each basis offset is determined as a function of the key.

1 21. (Original) The computer-readable storage medium of claim 19,
2 wherein generating the sequence of offsets involves determining an offset by
3 combining a base offset and a fixed offset, wherein the base offset is a function to

4 the key and the nonce, and the fixed offset is a function of the key and a position
5 of the fixed offset in a sequence of fixed offsets.

1 22. (Original) The computer-readable storage medium of claim 19,
2 wherein generating the sequence of offsets involves:
3 generating a sequence of fixed offsets from the key;
4 generating a base offset from the key and the nonce;
5 generating a sequence of translated offsets by combining each fixed offset
6 with the base offset to get a corresponding translated offset; and
7 using the sequence of translated offsets as the sequence of offsets.

1 23. (Original) The computer-readable storage medium of claim 22,
2 wherein the key determines a sequence of basis offsets and each fixed offset is
3 determined by xoring some combination of basis offsets.

1 24. (Original) The computer-readable storage medium of claim 23,
2 wherein each basis offset except for the first basis offset is determined by a shift
3 and a conditional xor applied to a previous basis offset.

1 25. (Original) The computer-readable storage medium of claim 24,
2 wherein the order that basis offsets are combined into fixed offsets is determined
3 according to a Gray code.

1 26. (Original) The computer-readable storage medium of claim 19,
2 wherein generating the sequence of offsets involves:
3 computing a sequence of basis offsets from the key;
4 computing a base offset from the key and the nonce; and

5 computing a sequence of translated offsets, wherein the first offset is
6 determined from the base offset, the key, and the nonce, and subsequent offsets
7 are determined by combining the prior translated offset with a basis offset.

1 27. (Original) The computer-readable storage medium of claim 19,
2 wherein generating the sequence of offsets involves:
3 computing a key-variant offset by enciphering a constant with the block
4 cipher, wherein the block cipher is keyed by a given key; and
5 computing the sequence of offsets using the key-variant offset.

1 28. (Original) The computer-readable storage medium of claim 19,
2 wherein computing the ciphertext body involves:
3 combining each message block in the message body with a corresponding
4 offset to produce a corresponding input block;
5 applying the block cipher to each input block to produce a corresponding
6 output block; and
7 combining each output block with a corresponding offset to produce a
8 corresponding ciphertext block.

1 29. (Original) The computer-readable storage medium of claim 19,
2 wherein computing the ciphertext fragment involves:
3 computing a precursor pad as a function of an offset;
4 computing a pad by applying the block cipher to the precursor pad; and
5 computing the ciphertext fragment by combining the message fragment
6 and the pad.

1 30. (Original) The computer-readable storage medium of claim 19,
2 wherein computing the tag involves:

3 computing a checksum as a function of the message and a sequence of
4 offsets; and
5 computing the tag as a function of the checksum, the key, and an offset.

1 31. (Original) The computer-readable storage medium of claim 19,
2 wherein computing the tag involves:
3 computing a checksum from the message blocks, the message fragment,
4 and a pad;
5 combining the checksum with an offset to produce a precursor full tag;
6 computing a full tag by applying the block cipher to the precursor full tag;
7 and
8 computing a tag as a portion of the full tag.

1 32. (Original) A computer-readable storage medium storing instructions
2 that when executed by a computer cause the computer to perform an
3 authenticated-encryption method that uses a key and a nonce to decrypt a
4 ciphertext into a message, the method comprising:
5 partitioning the ciphertext into a ciphertext body including a sequence of
6 n-bit ciphertext blocks, a ciphertext fragment of at most n bits, and a tag;
7 generating a sequence of offsets from the nonce and the key;
8 computing a message body using a block cipher, the ciphertext body, the
9 key, the nonce, and the sequence of offsets;
10 computing a message fragment using the block cipher, the ciphertext
11 fragment, the key, and an offset;
12 computing a new tag as a function of the message body; and
13 comparing the new tag with the tag;
14 if the new tag matches the tag, returning the message, wherein the message
15 includes the message body and the message fragment; and

16 otherwise, if the new tag does not match the tag, returning a message
17 invalid signal.

1 33. (Original) The computer-readable storage medium of claim 32,
2 wherein generating the sequence of offsets involves:
3 generating a sequence of fixed offsets from the key;
4 generating a base offset from the key and the nonce;
5 generating a sequence of translated offsets by combining each fixed offset
6 with the base offset to get a corresponding translated offset; and
7 using the sequence of translated offsets as the sequence of offsets.

1 34. (Original) The computer-readable storage medium of claim 32,
2 wherein computing the message body involves:
3 combining each ciphertext block in the ciphertext body with a
4 corresponding offset to produce a corresponding input block;
5 applying the block cipher to each input block to produce a corresponding
6 output block; and
7 combining each output block with a corresponding offset to produce a
8 corresponding message block.

1 35. (Original) The computer-readable storage medium of claim 32,
2 wherein computing the message fragment involves:
3 computing a precursor pad as a function of an offset;
4 computing a pad by applying the block cipher to the precursor pad; and
5 computing the message fragment by combining the ciphertext fragment
6 and the pad.

1 36. (Original) The computer-readable storage medium of claim 32,
2 wherein computing the tag involves:
3 computing a checksum as a function of the message body; and
4 computing the tag as a function of the checksum, the key, and an offset.

1 37. (Original) An authenticated-encryption apparatus that uses a key and a
2 nonce to encrypt a message into a ciphertext, the apparatus comprising:
3 a partitioning mechanism that is configured to partition the message into a
4 message body including a sequence of n-bit message blocks, and a message
5 fragment of at most n bits;
6 an offset generating mechanism that is configured to generate a sequence
7 of offsets from the nonce and the key;
8 an enciphering mechanism that is configured to compute a ciphertext body
9 using a block cipher, the message body, the key, the nonce, and the sequence of
10 offsets;
11 wherein the enciphering mechanism is additionally configured to compute
12 a ciphertext fragment using the block cipher, the message fragment, the key, and
13 an offset;
14 a tag computing mechanism that is configured to compute a tag as a
15 function of the message body, the message fragment, the sequence of offsets, and
16 the key; and
17 an assembly mechanism that is configured to define the ciphertext to
18 include the ciphertext body, the ciphertext fragment, and the tag.

1 38. (Original) An authenticated-encryption apparatus that uses a key and a
2 nonce to decrypt a ciphertext into a message, the apparatus comprising:

3 a partitioning mechanism that is configured to partition the ciphertext into
4 a ciphertext body including a sequence of n-bit ciphertext blocks, a ciphertext
5 fragment of at most n bits, and a tag;
6 an offset generating mechanism that is configured to generate a sequence
7 of offsets from the nonce and the key;
8 a deciphering mechanism that is configured to compute a message body
9 using a block cipher, the ciphertext body, the key, the nonce, and the sequence of
10 offsets;
11 wherein the deciphering mechanism is configured to compute a message
12 fragment using the block cipher, the ciphertext fragment, the key, and an offset;
13 a tag computing mechanism that is configured to compute a new tag as a
14 function of the message body; and
15 a comparison mechanism that is configured to compare the new tag with
16 the tag;
17 wherein if the new tag matches the tag, the apparatus is configured to
18 return the message, wherein the message includes the message body and the
19 message fragment; and
20 wherein if the new tag does not match the tag, the apparatus is configured
21 to return a message invalid signal.

1 39. (Original) An authenticated-encryption method that uses an n-bit block
2 cipher, a key, and an n-bit nonce to encrypt a message into a ciphertext, the
3 method comprising:
4 partitioning the message into m message blocks and one final fragment,
5 each message block having n bits and the final fragment having between 0 and n
6 bits;
7 using the block cipher, the key, and the nonce to generate a sequence of m
8 offsets, each offset having n bits;

9 using the block cipher, the key, the nonce, and the length of the message to
10 generate an n-bit final offset;
11 for each number i between 1 and m, xoring the i^{th} message block with the
12 i^{th} offset to determine an i^{th} input block;
13 for each number i between 1 and m, applying the block cipher, keyed by
14 the key, to the i^{th} input block, to determine an i^{th} output block;
15 for each number i between 1 and m, xoring the i^{th} output block with the i^{th}
16 offset to determine an i^{th} ciphertext block;
17 concatenating the m ciphertext blocks to determine a ciphertext body;
18 computing an encoded length by encoding the length of the final fragment
19 as an n-bit string;
20 xoring the encoded length with the final offset to determine a precursor
21 pad;
22 computing a pad by applying the block cipher, keyed by the key, to the
23 precursor pad;
24 xoring the final fragment with a portion of the pad to determine a
25 ciphertext fragment having the same length as the final fragment;
26 computing a padded ciphertext fragment by appending to the ciphertext
27 fragment a sufficient number of zero bits so that the padded ciphertext fragment
28 has n bits;
29 computing a checksum by xoring together the m message blocks, the pad,
30 and the padded ciphertext fragment;
31 computing a precursor full tag by xoring together the checksum and the
32 m^{th} offset;
33 determining a full tag by applying the block cipher, keyed by the key, to
34 the precursor full tag;
35 computing a tag as a portion of the full tag; and

36 defining the ciphertext to be the ciphertext body, the ciphertext fragment,
37 and the tag.

1 40. (Original) The method of claim 39, wherein the i^{th} offset from the
2 sequence of offsets is determined by:
3 computing a 0^{th} basis offset by applying the block cipher, keyed by the
4 key, to a constant;
5 for each positive number i , defining the i^{th} basis offset from the prior basis
6 offset by shifting the prior basis offset left one position, and then xoring the
7 resulting value with a constant that depends on the first bit of the prior basis
8 offset;
9 computing a base offset by applying the block cipher, keyed by the key, to
10 the xor of the 0^{th} basis offset and the nonce;
11 defining the first offset in the sequence of offsets as the xor of the 0^{th} basis
12 offset and the base offset; and
13 for each integer i greater than one, defining the i^{th} offset in the sequence of
14 offsets as the xor of the prior offset and the j^{th} basis offset, where j is the number
15 of zero-bits following the last one-bit when the number i is written in binary.

1 41. (Original) The method of claim 39, wherein the final offset is
2 determined by shifting the 0^{th} basis offset one position to the right, xoring a
3 constant that depends on the last bit of the 0^{th} basis offset, and then xoring the m^{th}
4 offset.

1 42-49 (Canceled).

1 50. (Original) A computer-readable storage medium storing instructions
2 that when executed by a computer cause the computer to perform an

3 authenticated-encryption method that uses an n-bit block cipher, a key, and an n-
4 bit nonce to encrypt a message into a ciphertext, the method comprising:
5 partitioning the message into m message blocks and one final fragment,
6 each message block having n bits and the final fragment having between 0 and n
7 bits;
8 using the block cipher, the key, and the nonce to generate a sequence of m
9 offsets, each offset having n bits;
10 using the block cipher, the key, the nonce, and the length of the message to
11 generate an n-bit final offset;
12 for each number i between 1 and m, xoring the i^{th} message block with the
13 i^{th} offset to determine an i^{th} input block;
14 for each number i between 1 and m, applying the block cipher, keyed by
15 the key, to the i^{th} input block, to determine an i^{th} output block;
16 for each number i between 1 and m, xoring the i^{th} output block with the i^{th}
17 offset to determine an i^{th} ciphertext block;
18 concatenating the m ciphertext blocks to determine a ciphertext body;
19 computing an encoded length by encoding the length of the final fragment
20 as an n-bit string;
21 xoring the encoded length with the final offset to determine a precursor
22 pad;
23 computing a pad by applying the block cipher, keyed by the key, to the
24 precursor pad;
25 xoring the final fragment with a portion of the pad to determine a
26 ciphertext fragment having the same length as the final fragment;
27 computing a padded ciphertext fragment by appending to the ciphertext
28 fragment a sufficient number of zero bits so that the padded ciphertext fragment
29 has n bits;

30 computing a checksum by xoring together the m message blocks, the pad,
31 and the padded ciphertext fragment;
32 computing a precursor full tag by xoring together the checksum and the
33 m^{th} offset;
34 determining a full tag by applying the block cipher, keyed by the key, to
35 the precursor full tag;
36 computing a tag as a portion of the full tag; and
37 defining the ciphertext to be the ciphertext body, the ciphertext fragment,
38 and the tag.

1 51. (Original) The computer-readable storage medium of claim 50,
2 wherein the i^{th} offset from the sequence of offsets is determined by:
3 computing a 0^{th} basis offset by applying the block cipher, keyed by the
4 key, to a constant;
5 for each positive number i, defining the i^{th} basis offset from the prior basis
6 offset by shifting the prior basis offset left one position, and then xoring the
7 resulting value with a constant that depends on the first bit of the prior basis
8 offset;
9 computing a base offset by applying the block cipher, keyed by the key, to
10 the xor of the 0^{th} basis offset and the nonce;
11 defining the first offset in the sequence of offsets as the xor of the 0^{th} basis
12 offset and the base offset; and
13 for each integer i greater than one, defining the i^{th} offset in the sequence of
14 offsets as the xor of the prior offset and the j^{th} basis offset, where j is the number
15 of zero-bits following the last one-bit when the number i is written in binary.

1 52. (Original) The computer-readable storage medium of claim 50,
2 wherein the final offset is determined by shifting the 0^{th} basis offset one position

3 to the right, xoring a constant that depends on the last bit of the 0th basis offset,
4 and then xoring the mth offset.

1 53-60 (Canceled).

1 61. (Original) An authenticated-encryption apparatus that is configured to
2 use an n-bit block cipher, a key, and an n-bit nonce to encrypt a message into a
3 ciphertext, comprising:

4 a partitioning mechanism that is configured to partition the message into m
5 message blocks and one final fragment, each message block having n bits and the
6 final fragment having between 0 and n bits;

7 an offset-generating mechanism that is configured to,

8 use the block cipher, the key, and the nonce to generate a
9 sequence of m offsets, each offset having n bits, and to

10 use the block cipher, the key, the nonce, and the length of
11 the message to generate an n-bit final offset;

12 an xoring mechanism, wherein for each number i between 1 and m, the
13 xoring mechanism is configured to xor the ith message block with the ith offset to
14 determine an ith input block;

15 an enciphering mechanism, wherein for each number i between 1 and m,
16 the enciphering mechanism is configured to apply the block cipher, keyed by the
17 key, to the ith input block, to determine an ith output block;

18 wherein for each number i between 1 and m, the xoring mechanism is
19 configured to xor the ith output block with the ith offset to determine an ith
20 ciphertext block;

21 a concatenating mechanism that is configured to concatenate the m
22 ciphertext blocks to determine a ciphertext body;

23 a computing mechanism that is configured to compute an encoded length
24 by encoding the length of the final fragment as an n-bit string;
25 wherein the xoring mechanism is configured to xor the encoded length
26 with the final offset to determine a precursor pad;
27 wherein the computing mechanism is configured to compute a pad by
28 applying the block cipher, keyed by the key, to the precursor pad;
29 wherein the xoring mechanism is configured to xor the final fragment with
30 a portion of the pad to determine a ciphertext fragment having the same length as
31 the final fragment;
32 wherein the computing mechanism is configured to compute a padded
33 ciphertext fragment by appending to the ciphertext fragment a sufficient number
34 of zero bits so that the padded ciphertext fragment has n bits;
35 wherein the computing mechanism is configured to compute a checksum
36 by xoring together the m message blocks, the pad, and the padded ciphertext
37 fragment;
38 wherein the computing mechanism is configured to compute a precursor
39 full tag by xoring together the checksum and the mth offset;
40 wherein the computing mechanism is configured to determine a full tag by
41 applying the block cipher, keyed by the key, to the precursor full tag;
42 wherein the computing mechanism is configured to compute a tag as a
43 portion of the full tag; and
44 a defining mechanism that is configured to define the ciphertext to be the
45 ciphertext body, the ciphertext fragment, and the tag.

1 62-66 (Canceled)